

Generarea semnalelor standard, de test

1 Scopul lucrării

Familiarizarea cu modul de generare și reprezentare în mediul Matlab a semnalelor de test, considerate standard în ingineria sistemelor automate.

Însușirea noțiunilor de utilizare a echipamentelor necesare obținerii practice a semnalelor de testare a sistemelor continue, liniare invariante în timp (LTI).

2 Considerații teoretice

Semnalele continue în timp, considerate standard pentru testarea sistemelor continue, LTI sunt:

- semnalul impuls (Dirac), $\delta(t) = \begin{cases} \infty, & t = 0 \\ 0, & t \neq 0 \end{cases}$
- semnalul treaptă unitate, $1(t) = \begin{cases} 0, & t < 0 \\ 1, & t \geq 0 \end{cases}$
- semnalul rampă, $r(t) = \begin{cases} 0, & t < 0 \\ t, & t \geq 0 \end{cases}$
- semnalul sinusoidal $u(t) = \sin(\omega t)$, $\omega \in \mathbb{R}_+$, $t \in \mathbb{R}$, cu ω notație pentru pulsația semnalului.

Prin respectarea considerațiilor asupra modului de referire al sistemelor dinamice sub formă de diagrame bloc (Figura 2.1) sau ca un set de semnale de intrare / ieșire, notația $u(t)$ va fi utilizată pentru referirea semnalului de intrare, respectiv $y(t)$ este folosită pentru referirea semnalului de ieșire. Se renunță la modul de referire preluat din matematică și care indică prin notație forma semnalului, i.e. $\delta(t)$, $1(t)$, $r(t)$, etc.

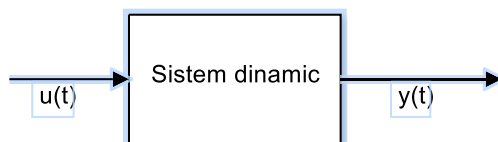


Figura 2.1 Reprezentarea prin diagrame bloc

Pentru simularea comportamentului unui sistem LTI la diferite semnale de intrare, se consideră ca domeniul de interes doar domeniul pozitiv de variație al timpului. Această limitare nu neglijează comportamentul anterior al sistemului ci stabilește ca moment inițial de simulare, momentul $t=0$, nivelul de energie înmagazinată de sistem fiind precizat prin condițiile inițiale nenule.

Mediul de simulare Matlab rulează pe un echipament de natură digitală. Perceperea semnalelor ca semnale analogice (continue în timp și amplitudine) fiind realizată printr-un artificiu de reprezentare grafică a valorilor discrete. În mod implicit reprezentarea grafică

Lucrarea nr. 1

se realizează prin unirea cu linie continuă a valorilor reprezentate. O atenție deosebită trebuie acordată modului în care se stabilește pasul cu care este eșantionat timpul și modului în care se interpretează (citesc) comentariile implicite de pe axele reprezentărilor grafice.

2.1 Generarea semnalelor în mediul Matlab

Mediul de simulare Matlab oferă multiple variante de a genera și reprezenta grafic un semnal analogic [2]. Pentru oricare variantă, se recomandă inițializarea unui vector de timp, în care pasul de generare să se aleagă ținând cont de limitările procesului sau sistemului care va fi studiat și de caracteristicile calculatorului pe care rulează mediul de simulare Matlab.

Semnalul impuls

Sunt prezentate variantele agreeate în analiza sistemelor LTI, de a genera un semnal impuls unitar. Primul exemplu reprezintă o variantă compactă de a genera un vector cu valori reprezentate în dublă precizie. Se apelează la operațiile binare disponibile (\geq), împărțirea cu o valoare reală convertind valorile binare în reale.

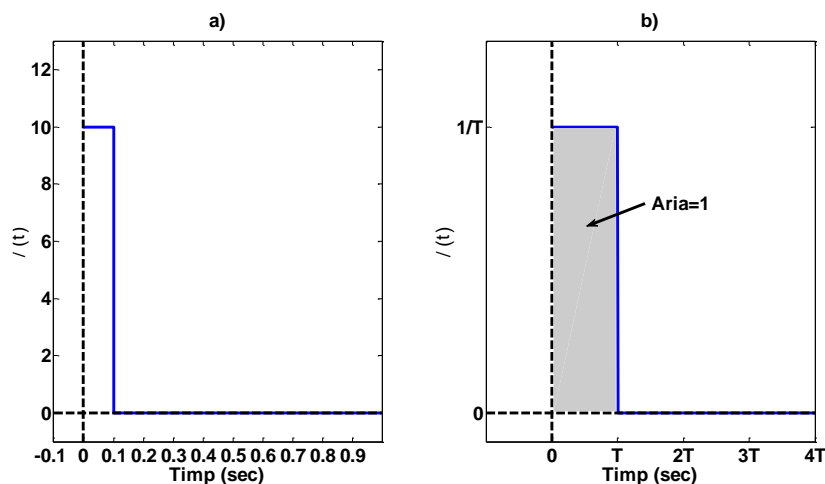


Figura 2.2 Semnalul impuls unitate (a). valori concrete, b). valori abstracte, $T=0.1$ sec.)

Pentru obținerea reprezentării semnalului impuls unitate din Figura 2.2, s-a utilizat următorul script în Matlab:

```
T=0.1;% constata de timp a sistemului
n=10;t=0:T/100:n*T;% timpul de simulare
u=(t<=T)/T;% semnal impuls unitate
subplot(121); % 2 subgrafice pe linie
plot(t,u,'LineWidth',2);% reprezentare grafica a semnalului u
axis([-0.1 n*T -1 13]); % setare suprafata de reprezentare
```

```
hold; % retine reprezentarea grafica din fereastra activa
plot([0 0],[-1 13],'--k',[-0.1 n*T],[0 0],'--k');
hold; % se renunta la mentinerea reprezentarilor curente
xlabel('Timp (sec)','FontSize',12); % comentariu pe abscisa
% comentariu ordonata
ylabel('$\delta(t)$','FontSize',14,'interpreter','Latex');

subplot(122); % a doua subfereastra activa
% hașurare suprafață
area([0 0.1],[10 10],'FaceColor',[.5 .5 0.5]);
plot(t,u,'LineWidth',2);
axis([-0.1 4*T -1 13])
hold;plot([0 0],[-1 13],'--k',[-0.1 n*T],[0 0],'--k');
% precizarea punctelor active de pe abscisa
set(gca,'XTick',[0:T:4*T,9*T]);
% textul punctelor de pe abscisa
set(gca,'XTickLabel',{'0','T','2T','3T','4T','9T'})
% precizarea punctelor active de pe ordonata
set(gca,'YTick',[0 1/T]);
% denumirea punctelor considerate pe ordonata
set(gca,'YTickLabel',{'0','1/T'});
% plasare cu mouse-ul de text pe reprezentarea grafica
gtext('Aria=1','FontSize',12);
xlabel('Timp (sec)','FontSize',12);
ylabel('$\delta(t)$','FontSize',14,'interpreter','Latex')
```

Dacă se apelează la lucrul cu matrice (vectori), se obține o variantă de cod accesibilă din punct de vedere a estimării rezultatului execuției fiecărei comenzi. Prin scriptul următor se poate însuși modul de generare și referire a vectorilor în mediul Matlab:

```
T=0.1;% constata de timp a sistemului
n=4;t=0:T/100:n*T;% timpul de simulare
l=length(t);% lungimea vectorului de timp t
% generare vector linie, de lungime l, cu toate elementele 0
u=zeros(1,l);% generare vector linie, de lungime l
u(1:100)=1/T;% modificarea primelor 100 de valori din u
plot([0 0],[-1 13],'--k',[-0.1 n*T],[0 0],'--k',t,u,'b*');
set(gca,'XTick',[0, T]);
set(gca,'XTickLabel',{'0','T'})
set(gca,'YTick',[0 1/T]);
set(gca,'YTickLabel',{'0','1/T'})
xlabel('Timp (T)')
ylabel('$\delta(t)$','interpreter','Latex')
title('Semnal impuls unitate');% comentariu titlu
grid; % trasare linii de control
```

Lucrarea nr. 1

În Figura 2.3 este prezentat graficul obținut în urma executării comenzilor precedente. Spre deosebire de graficele din Figura 2.2, în acest caz semnalul este reprezentat prin puncte, neunite prin linie continuă.

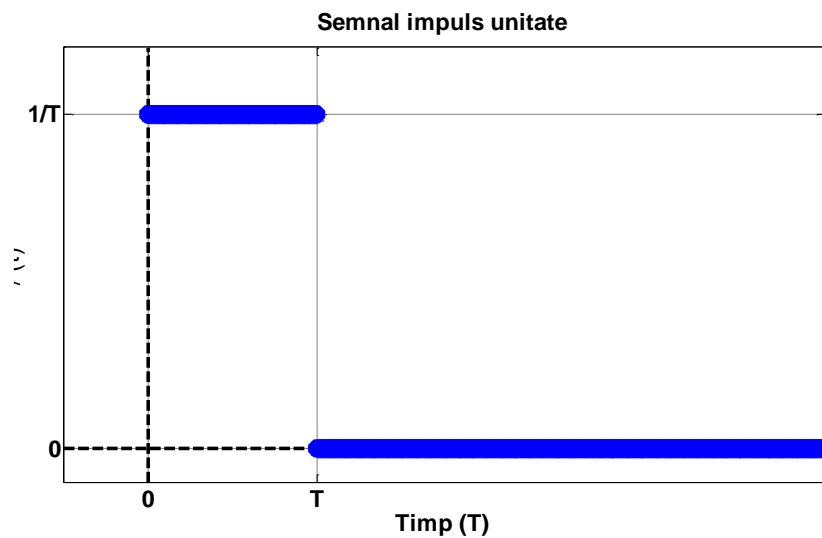


Figura 2.3 Semnal impuls unitate reprezentat prin puncte ($T=0.1$ sec.)

Ambele scripturi utilizează același vector de timp, eșantionat la $T/100$ ($1ms$). Pentru alte valori ale eșantionului de timp, se obțin diferite reprezentări grafice în Figura 2.4.

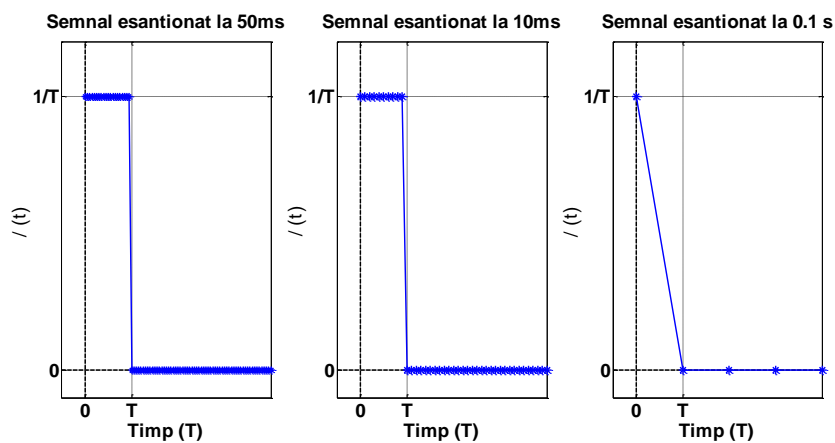


Figura 2.4 Semnale generate cu diferite perioade de eșantionare ($T=0.1$ sec.)

Semnalul treaptă unitate

Pentru generarea semnalului treaptă unitate se pot refolosi variantele de script precedente, modificările esențiale fiind la nivelul comenzii de declarare a semnalului u . Observațiile legate de modul de generare al vectorului de timp, cu un pas de generare automată (perioadă de eșantionare) direct legat de aplicația în care va fi utilizat semnalul de intrare u . Scriptul următor prezintă și modul de implementare a unei bucle cu comenzi din Matlab. Rezultatul execuției comenzilor din scriptul următor este graficul din Figura 2.5, cu trei valori considerate pentru pasul de generare automată a semnalului u .

```
T=0.1;n=4;e=[T/100 T/10 T];
for i=1:length(e)
    t=0:e(i):n*T;l=length(t);
    % generare vector linie, de lungime l, cu toate
    elementele 1
    u=ones(1,l);
    subplot(1,3,i)
    plot(t,u,'b*-');
    set(gca,'XTick',[0:T:n*T]);
    set(gca,'YTick',[0 0.5 1]);
    set(gca,'YTickLabel',{'0','0.5','1'})
    xlabel('Timp (s)');ylabel('u(t)');
    title(['Semnal esantionat la ',num2str(e(i)),'s']);
    axis([-0.03 n*T -0.1 1.2]);grid
end
```

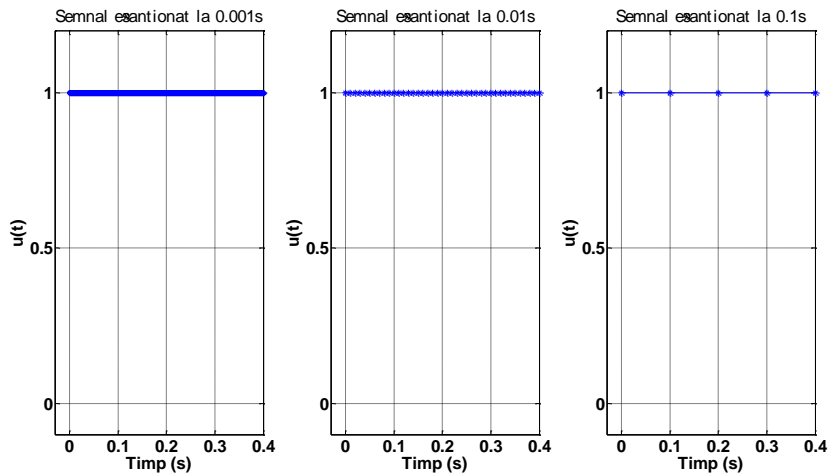


Figura 2.5 Semnal treaptă unitate

Lucrarea nr. 1

Semnalul rampă unitate

Plecând de la expresia matematică a semnalului rampă de pantă $\arctg(a)$:

$$u(t) = a \cdot t, \text{ cu } a \in \mathbb{R} \quad (1)$$

se poate utiliza următorul script pentru a genera și reprezenta grafic (Figura 2.6) un semnal rampă de diferite valori ale pantei:

```
t=0:0.01:0.1;a=[sqrt(3)/3 1 sqrt(3)];
plot(t,a(1)*t,'kv-',t,a(2)*t,'k>-',t,a(3)*t,'k<-');
legend('$a=\frac{\sqrt{3}}{3}$','$a=1','$a=\sqrt{3}$',...
'interpreter','Latex');
title('Semnale rampă u{a}','interpreter','Latex');
grid; xlabel('Timp (sec.)');ylabel('u(t)=a*t')
poz=[0.02 0.06 0.10];
for i=1:3
    panta=atan(a(i))*180/pi;
    text(0.07,poz(i), ['panta de ', num2str(panta), '^{\circ}']);
```

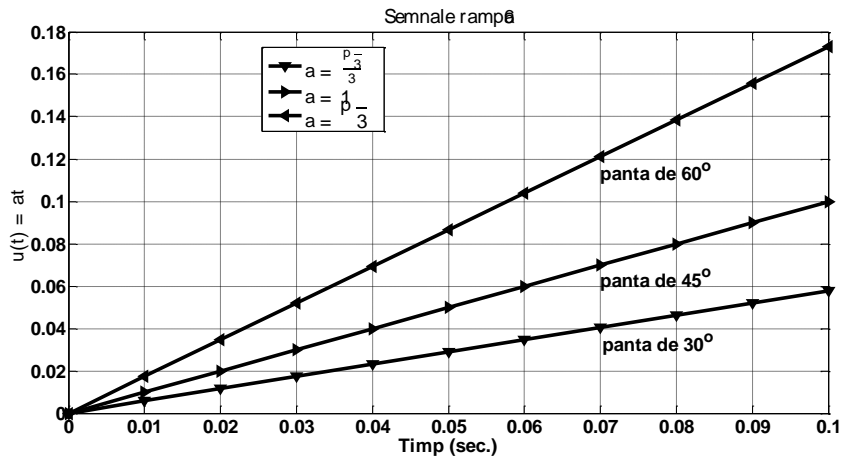


Figura 2.6 Exemple de semnal rampă de diferite pante

Semnalul sinusoidal

Forma generală a unei sinusoide este determinată de trei parametri: amplitudine notată cu A , pulsație notată cu ω și defazaj notat cu φ :

$$u(t) = A \cdot \sin(\omega \cdot t + \varphi) \text{ cu } A, \omega \in \mathbb{R}^+ \quad (2)$$

sau evidențiind frecvența f :

$$u(t) = A \cdot \sin\left(\underbrace{2 \cdot \pi \cdot f}_{\omega} \cdot t + \varphi\right) \text{ cu } f \in \mathbb{R}^+ \quad (3)$$

De interes pentru aplicațiile teoretice este cazul particular al sinusoidei de amplitudine 1 și nedefazată. Se preferă utilizarea pulsației deoarece aceasta se regăsește în modelul matematic care va fi utilizat pentru analiza sistemelor dinamice. Al doilea aspect care înclină spre utilizarea pulsației și nu a frecvenței este sintaxa din Matlab a funcției `sin` prin intermediul căreia se va genera semnalul sinusoidal.

Pentru reprezentarea grafică a unui semnal sinusoidal de pulsație precizată (Figura 2.7), se propune următoarea variantă de script:

```
w=2;% pulsația semnalului de intrare
T=2*pi/w;% perioada semnalului
n=30;% numar de esantioane pe o perioada
nT=6; % numar de perioade de reprezentat
t=0:T/n:nT*T; % timp de reprezentare
y=sin(w*t); % semnal sinusoidal
plot(t, y, 'LineWidth',2); % reprezentarea grafica
title(['Semnal sinusoidal \it{sin()}',num2str(w),'t'])
axis([-0.1 nT*T -1.2 1.2]);grid
set(gca,'XTick',[0:T:nT*T]);
set(gca,'YTick',[-1 0 1]);
set(gca,'YTickLabel',{'-1','0','1'})
xlabel('Timp (sec.)');ylabel('Amplitudine')
```

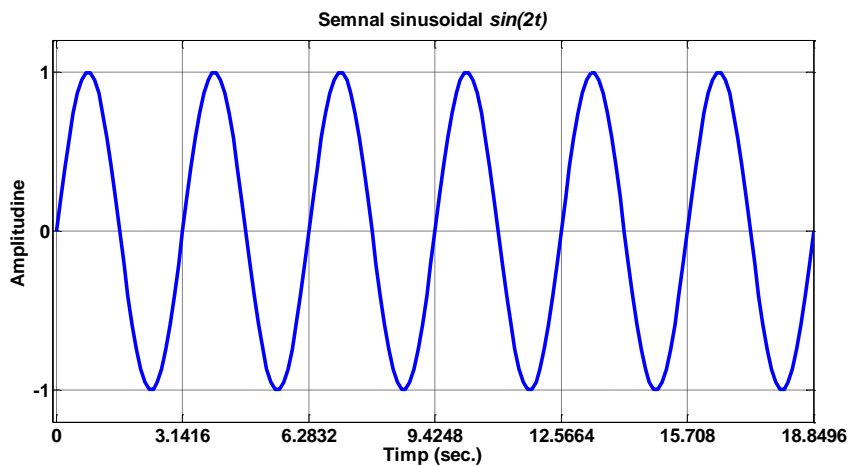


Figura 2.7 Semnale sinusoidal de pulsație 2 rad/s, reprezentat pe 6 perioade

Semnalul dreptunghiular

Direct legat de aplicațiile practice, semnalul dreptunghiular este o facilități comună a generatoarelor de funcții. Din punct de vedere matematic, este un semnal compus, studiul detaliat fiind realizat în cadrul altor materii de specialitate. Modul de declarare al unui semnal dreptunghiular în mediul de simulare Matlab se poate realiza prin funcția `square`, similară ca și sintaxă cu funcția `sin`. Pentru exemplificare, în Figura 2.8 este prezentat

Lucrarea nr. 1

semnalul dreptunghiular obținut utilizând scriptul precedent, în care funcția `sin` s-a înlocuit cu `square`.

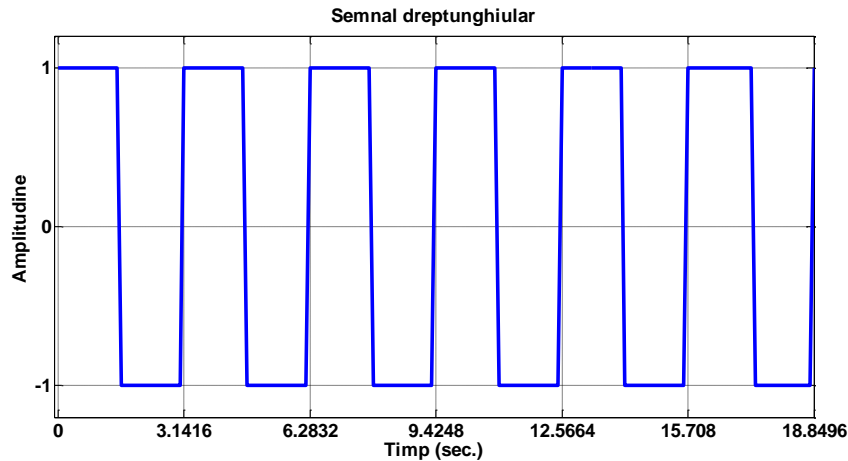


Figura 2.8 Semnal dreptunghiular de pulsație 2 rad/s, reprezentat pe 6 perioade

3 Probleme

3.1 Scripturi în Matlab

1. Să se genereze și să se reprezinte grafic un semnal sinusoidal de frecvență $f=50\text{Hz}$, pe durata a două perioade.
2. Să se genereze un semnal dreptunghiular cu cele două nivele în $+1.5$, respectiv -0.7 , de frecvență $f=100\text{ Hz}$.
3. Să se explice erorile care apar la rularea următorului script; să se corecteze sau să se modifice comenzile astfel încât să se obțină reprezentarea din Figura 3.1

```
t=0:0,1:1;  
u1=sin(4t);u2=2*t;u3=u1*u2;  
plot(u3,t,u2,t);  
grind; titel('Un sinus cu amplitudinea liniara in timp');  
xlabel('Timp (sec.)');ylabel('Amplitudine ')
```

4. Să se realizeze un fișier de tip funcție care să genereze un semnal dreptunghiular similar funcției `square`; să se utilizeze doar atribuiri și operații cu vectori; parametrii de intrare ai funcției vor fi timpul de simulare dat sub formă de vectori cu valori pozitive și monotone și pulsația semnalului sub formă de număr real și pozitiv; parametru de ieșire se dorește de forma unui vector de aceeași lungime cu parametrul de intrare timp și care să modeleze pe durata precizată un semnal dreptunghiular cu cele două nivele în $+1$ respectiv -1 ; modul de apelare să fie de forma:

```
u=nume_functie(t,w)
```

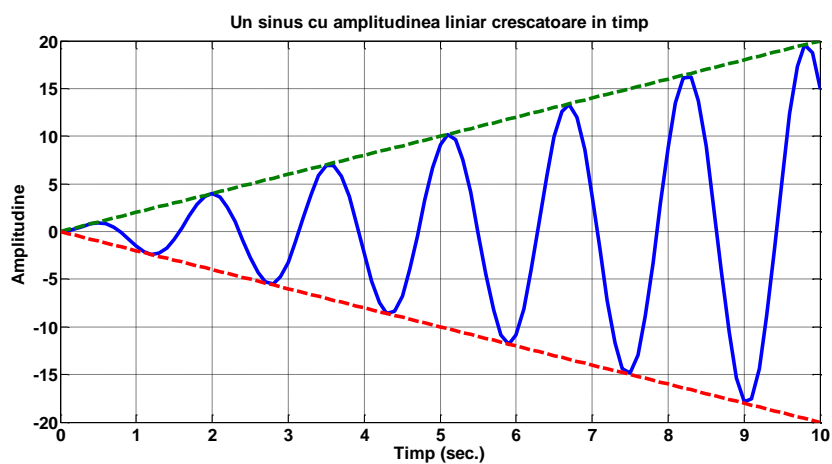



Figura 3.1 Sinus de amplitudine variabilă (rezonanța liniară în timp)

